DOCUMENT RESUME

ED 386 920 FL 023 025

AUTHOR Pavanini, Pietro
TITLE Errori 'programmati' ('Programmed' Errors).
PUB DATE 93
NOTE 8p.
AVAILABLE FROM La Nuova Italia Editrice, Scandicci (Firenze), Italy.
PUB TYPE Journal Articles (080) -- Viewpoints (Opinion/Position Papers, Essays, etc.) (120)
LANGUAGE Italian
JOURNAL CIT Italiano e Oltre; v8 p100-106 1993

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Cloze Procedure; *Computer Assisted Instruction; *Dictation; English (Second Language); Error Analysis (Language); Foreign Countries; *Italian; Listening Comprehension; *Second Language Instruction; Second Language Learning; *Songs
IDENTIFIERS *Belli Capelli

ABSTRACT

This article begins by affirming that the use of computers in language teaching makes sense only if teachers have computer expertise that allows them to create programs that fit their specific pedagogical needs. A program is presented called "musical dictation," which serves as an example of what can be done by a teacher with easily-acquired computer skills. Students listen to a song and are given the text with words missing. This exercise differs from computerized cloze exercises and from Storyboard exercises in two ways: (1) students do not guess words that will eventually appear but they fill in, as they listen, words that they have already seen on the monitor; and (2) the comments that the program makes in response to the students' answers are not limited to 'correct' or 'incorrect' but offer additional help in understanding errors. The article then describes a program designed for students of Italian as a Second Language based on the song "Belli Capelli." Possible student responses are analyzed and the feedback offered by the program is demonstrated. The article contains a bibliography with titles of CALL bibliographies, books on CALL, CALL software suppliers, CALL periodicals, and CALL software. (CFM)

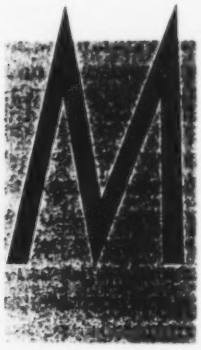
****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Errori 'programmati'

ED 386 920

100

1
COMPUTER ASSISTED LANGUAGE LEARNING



olto si è detto e scritto sull'informatica e le sue applicazioni linguistiche; in particolare, la bibliografia relativa al tema computer e glottodidattica ha raggiunto proporzioni tali da indurre Udo O. H. Jung ad aggiornare il già denso repertorio bibliografico apparso nel 1988. La sigla CALL (*Computer Assisted Language Learning*) non dovrebbe insomma più apparire, almeno agli insegnanti di lingue, come uno fra i tanti misteriosi acronimi.

Sono usciti infatti diversi manuali pratici che si rivolgono anche ai non esperti e, con la loro intenzionale semplicità, mirano a sgombrare il campo con chiarezza su quel che il computer, almeno nel campo delle lingue straniere, può e quel che non può fare.

Altrettanto ricca è stata negli ultimi anni la produzione di *software* per l'insegnamento delle lingue straniere. Si va dai semplici programmi tutoriali ed esercitativi ai cosiddetti «programmi autore», molto apprezzati perché consentono all'insegnante di inserire i propri materiali didattici all'interno di una cornice fornite dal programma. In *da parola a parola* (*storyboard* nell'originale inglese), per esempio, forse il programma autore più noto fra gli utenti di *software* CALL, ogni parola del testo viene tradotta in una successione di quadratini, uno per ogni lettera. Gli allievi ricostruiscono il testo parola per parola; la parola 'indovinata' apparirà tutte le volte che ricorre nel testo. Man mano la struttura del testo svelato consentirà, come in una sorta di *puzzle*, ipotesi sempre più plausibili sulle parole mancanti e sul contesto. Il programma tiene conto dei tentativi effettuati, degli eventuali aiuti richiesti e del tempo impiegato per risolvere l'esercizio.

E tuttavia, «l'euforia dei primi anni da pionieri» sembra aver ceduto il posto a un più sobrio ripensamento. «L'iniziale entusiasmo per il microcomputer», come sussidio nell'apprendimento delle lingue straniere, «sembra essersi ampiamente perduto causa delle limitate forme di software» oggi disponibili[1]. «L'insegnamento delle lingue assistito dal computer in una strada senza uscita?», si chiede di conseguenza Gerhard Wazel, e imputa proprio alla facilità d'impiego e di manipolazione degli stessi programmi autore, e al fatto che questi non richiedano praticamente nessuna conoscenza tecnica di sistemi e linguaggi di programmazione, la bassa qualità dei materiali CALL oggi in commercio: «Le esigue difficoltà nella produzione del programma [...] si pagano con una concezione didattica decisamente ristretta.»[2]

PIETRO PAVANINI

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Applicazioni informatiche per l'insegnamento dell'italiano L2

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Pietro Pavanini
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

ITALIANO E OLTRE, VIII (1993), pp. 100-106

FL023025

BEST COPY AVAILABLE

Un impiego didatticamente produttivo del computer nell'educazione linguistica ha dunque senso solo se l'insegnante possiede le conoscenze professionali che gli consentono di sfruttare in pieno le potenzialità della macchina e di costruire i propri programmi adattandoli alle esigenze, al livello di apprendimento, alle difficoltà tipiche del suo gruppo di allievi.

Questo non significa che l'insegnante di lingue debba obbligatoriamente trasformarsi in un esperto di informatica. Se per la creazione di seri programmi CALL la soluzione ideale resta ovviamente quella della collaborazione fra glottodidatta e tecnico programmatore – ma la scarsa praticabilità di un tal genere di soluzione, almeno a livello scolastico, è facilmente intuibile –, nondimeno, a dare un impulso più professionale all'impiego di materiali CALL, basterebbe un aggiornamento degli insegnanti meno dispersivo e più mirato. Meritano di essere qui ricordate alcune iniziative di aggiornamento che, da alcuni anni, si muovono in questa direzione: i «Corsi di perfezionamento in informatica nella didattica», organizzati dal Dipartimento di Scienze dell'Educazione dell'Università di Firenze e i «Corsi di perfezionamento in glottodidattica con tecnologie avanzate» presso il Centro Linguistico Interfacoltà dell'Università di Venezia.

2
DETTATO CANORO

Il piccolo programma che qui proponiamo[3] vuole essere una dimostrazione di come la sintassi di programmazione per la gestione di un database possa, con relativa semplicità e con alcune conoscenze tecniche facilmente apprendibili, trovare applicazione in un'esercitazione linguistica.

L'esercizio, diretto nel nostro caso a studenti di italiano come lingua straniera, è una vecchia conoscenza degli utenti del laboratorio linguistico; può trovare tuttavia applicazione anche con il semplice impiego di un registratore a cassette. Si tratta di ascoltare più volte una canzone, di cui ciascun allievo ha sottomano il testo scritto ma incompleto, e di integrarla con le parti mancanti. Una variante più complessa, applicabile solo al laboratorio linguistico, prevede la trascrizione dell'intero testo. In tal caso l'insegnante trasferisce la registrazione della canzone della *console* alle cabine o ai banchi degli allievi che lavorano poi autonomamente, ciascuno manovrando a piacere il proprio registratore, entro un tempo prestabilito.

Dunque una sorta di esercizio *cloze* non sistematizzato che, in forma più leggera, tende a sviluppare e/o testare alcune abilità integrate – comprensione auditiva, discriminazione fonetico-fonologica e correttezza ortografica – sulla base di un materiale autentico che facilmente suscita interesse e divertimento fra un pubblico di stranieri. L'ascolto e lo sfruttamento dei testi di canzoni si presta, fra l'altro, ottimamente all'insegnamento delle lingue e, con opportuna preparazione, rappresenta un'attività che va ben oltre quella di un intervallo distensivo destinato all'ultima ora del corso. Il successo e i buoni risultati ottenuti hanno visto crescere l'interesse dei ricercatori, e la pubblicazione di raccolte ricche di spunti anche divertenti e di utili suggerimenti hanno rivalutato l'importanza di questa attività 'minore' nell'ambito glottodidattico[4].

La novità della versione al computer di un tal genere di esercizio consiste essenzialmente nella possibilità di predisporre un'analisi più approfondita delle singole risposte (errate) e di fornire all'allievo un commento discriminante per ciascuna di esse che lo aiuti, in un successivo ascolto della canzone, ad autocorreggersi o a individuare le parole che non ha compreso.

Due importanti caratteristiche contraddistinguono quindi questo esercizio rispetto ai *cloze* computerizzati e alla ricostruzione di testi sul modello di *Storyboard*:

(1) Non si tratta di indovinare parole o *items* eventualmente presenti in un testo del tutto o parzialmente nascosto, ma di completare, ascoltando, un contesto quasi integralmente già visualizzato (sul monitor).

(2) I commenti che il programma fornisce alle ri-

sposte degli allievi non si limitano alla semplice differenziazione giusto/sbagliato, ma distinguono gli errori più comuni, offrendo aiuti mirati e indicazioni utili a correggerli. Il limite di tanti programmi autore in commercio sta proprio nella loro schematicità nell'analisi delle risposte degli allievi; essi non distinguono «la *correttezza* linguistica della *appropriatezza*, l'errore di distrazione da quello per ignoranza (*error / mistake*).» [5]

Ovviamente, non potendo né prevedere né inserire nel programma tutte le variabili di risposte, ci si limiterà a memorizzare quelle che, per esperienza, hanno un maggior tasso di frequenza. Il criterio di scelta sia della canzone sia, soprattutto, delle parole mancanti, terrà quindi conto del livello di apprendimento e delle difficoltà statisticamente riscontrate nel gruppo a cui ci si rivolge. Sarà inoltre opportuno che le parole da inserire siano a una certa distanza l'una dall'altra, affinché l'allievo abbia il tempo di batterle mentre prosegue l'ascolto della canzone.

La canzone scelta per il nostro modello è un testo di Francesco De Gregori, *Belli capelli*. La melodia lenta, la brevità del brano (ciò che, fra l'altro, permette di visualizzarlo interamente sul monitor), il ricorso a rime o assonanze e allitterazioni, la semplicità e la ripetitività del lessico la rendono adatta a una classe di principianti/intermedi. Le difficoltà sono dovute essenzialmente alla presenza di certe licenze poetiche e ad alcuni raddoppiamenti sintattici e *liasons* tipici della parlata romana.

Dopo aver lanciato il programma, l'allievo viene invitato a scrivere il proprio nome; appare quindi il testo della canzone con le parole mancanti:

Belli capelli, capelli neri, (r. 1)
che t'ho ... tutta la notte (r. 2)
e tu ... dov'eri; (r. 3)
capelli ... che arrivavano fino al mare, (r. 4)
belli capelli che nessuno li può ... (r. 5)
Belli capelli, capelli ... (r. 6)
che in mezzo a tutta quella ... mi sentivo solo; (r. 7)
capelli ... che sei partita e chi lo sa se torni, (r. 8)
belli capelli che mi tradivano tutti i ... (r. 9)
Capelli come autostrade la mattina sopra il tuo ..., (r. 10)
e quando tira vento diventano i capelli di un ..., (r. 11)
capelli così lontani che nessuno li può vedere, (r. 12)
capelli così ... che basta niente che li fai cadere, (r. 13)
Belli capelli, capelli ..., (r. 14)
che si fermarono a una fontana a ... gli anni, (r. 15)
capelli stanchi dentro allo ... di un bicchiere di vino, (r. 16)
belli capelli che stanotte è notte m,a verrà mattino, (r. 17)

Ascoltando la canzone da un registratore a cassette gli allievi ne completano il testo battendo sulla tastiera le tredici parole mancanti[6] (premendo «invio» il cursore si posiziona automaticamente all'inizio della successiva parola da inserire). Per consentire allo studente una maggiore 'libertà' d'errore, l'ampiezza degli spazi vuoti supera mediamente di una/due battute la reale lunghezza della parola; per lo stesso motivo si è rinunciato a tradurre in quadratini, o altri simboli grafici, le singole lettere mancanti.

Al termine dell'ascolto appare sullo schermo un messaggio con le istruzioni per il controllo delle risposte:

Portati con il cursore sopra le singole risposte e premi F1 per vedere gli errori o per avere suggerimenti

I messaggi che appariranno di volta in volta sul monitor diranno all'allievo se la parola battuta è esatta o gli forniranno indicazioni sul tipo di errore eventualmente commesso. Terminato il controllo, gli allievi riascoltano la canzone e cercano di autocorreggersi con l'aiuto dei suggerimenti precedentemente ottenuti. Per facilitarsi nella correzione, sullo schermo vengono automaticamente evidenziate in rosso le parole errate e in bianco su sfondo grigio quelle esatte. Segue quindi un ulte-

riore controllo e, se l'insegnante lo ritiene opportuno, un nuovo ciclo di ascolto. La stampa del testo completo e corretto conclude l'esercizio.

Il programma memorizza le risposte battute dagli allievi e le archivia automaticamente in un apposito file (*Belli. Err*, nel nostro caso) con il nome dello studente, distinguendo fra quelle del primo e quelle – eventualmente corrette – del secondo ascolto. L'insegnante ha così a disposizione un utile strumento sia di valutazione sia, soprattutto, di rilevamento statistico delle più frequenti tipologie di errore, ciò che gli consente di impostare con maggiore accuratezza il programma di analisi delle risposte.

3
Analisi delle risposte

Per l'analisi delle risposte fornite dagli allievi, indubbiamente la parte più interessante dell'esercizio, abbiamo utilizzato il linguaggio di programmazione per la gestione di database *Clipper*. La sua sintassi consente infatti l'inserimento di diverse variabili di risposte e conseguentemente di discriminare un'ampia gamma di possibili errori.

Prima di inserire le stringhe del programma si è perciò provveduto a sistematizzare le probabilità d'errore, prevedendo due categorie variabili:

(1) variabili comuni a tutte le parole

(2) variabili specifiche per le singole parole.

Le variabili del gruppo (1) sono state a loro volta così suddivise:

1. 1. La parola è esatta

1. 2. Non viene inserita nessuna parola

1.3. La prima o l'ultima lettera della parola non sono esatte

1.4. Manca la prima o l'ultima lettera della parola

1.5. La parola 'assomiglia' a quella esatta

1.6. La parola è del tutto errata

Per quanto riguarda le variabili del gruppo (2) si è tenuto conto della tipologia di errori più comuni fra gli stranieri a seconda della loro lingua madre:

2.1. Discriminazione fra consonanti singole e doppie

2.2. Discriminazione dei fonemi/grafemi tipici dell'italiano: *ci / œ - chi / che*; *cu / qu*; *gi / ge - ghi / ghe*; *gli*; *gn*; *s-z*; *sci / sce*.

2.3. Discriminazione fra consonanti sorde e sonore: *p-b*; *t-d* ecc.

2.4. Concordanza nome/aggettivo

2.5. Suffissi di forme verbali *are; ato*, ecc.

2.6. Accenti e apostrofi

Vediamo ora come appaiono le stringhe del programma relative alla prima parola da inserire («aspettato» (r. 2), con l'indicazione delle variabili ([var.]) di cui si è tenuto conto nell'analisi delle risposte possibili:

if asp = «aspettato»
risp = «esatto» (var. 1.1.)
elseif empty (asp)
risp. «È un sinonimo di "atteso"» (var. 1.2.)
elseif asp = «spettato»
risp = «Manca una lettera all'inizio della parola» (var. 1.4.)
elseif asp = «aspettat»
risp = «Manca una lettera alla fine della parola» (var. 1.4.)
elseif asp = «spetato»
risp = «Manca una lettera all'inizio della parola e attento alle doppie consonanti» (var. 1.4.) (var. 2.1)
elseif asp = «aspetato». or.asp= «aspettatto». or.asp = «aspetatto»
risp = «Attento alle doppie consonanti» (var. 2.1.)
elseif right (rtrim(asp), 3) «ato», and. left. (rtrim(asp), 6) = «aspett» (a)
risp = «L'ausiliare ho *ti aiuta a capire che si tratta di un participio passato»* (var. 2.5)
elseif soundex (asp) = soundex («aspettato») (b)
risp = «La parola assomiglia solo a quella esatta» else (var. 1.5.)
else
risp = «Mi dispiace, non è questa la parola» (var. 1.6)
endif

Traducendo dall'inglese un po' maccheronico tipico dei linguaggi di programmazione, le prime

stringhe significano grosso modo:«se [*if*] la parola battuta nel primo spazio [*asp*] è [=] «aspettato» il commento che appare in video [*risp* =] sarà «Esatto»; «se invece [*elseif*] lo spazio rimane vuoto [*empty (asp)*]» il commento sarà «è un sinonimo di "atteso"». E così via fino all'ultima stringa: «in tutti gli altri casi» [*else*] il commento sarà «Mi dispiace, non è questa la parola». La stringa [a] significa: «se le ultime tre lettere della parola [*right (rtrim(asp), 3*] sono diverse [<>] da «ato» e [*.and.*] le prime sei sono uguali a [*left(rtrim(asp), 6)* =] «aspett» il commento sarà «L'ausiliare *ho* ti aiuta a capire che si tratta di un participio passato». La stringa [b] si riferisce alla variabile 1.5.: si tratta di un algoritmo in grado di riconoscere – rapportando la codifica scritta alla pronuncia parole – foneticamente simili all'originale. Tale algoritmo è stato qui introdotto solo in via sperimentale: la base fonologica di raffronto su cui esso si basa è infatti quella dell'anglo-americano, il che rende ovviamente possibili incongruenze rispetto all'italiano . E' tuttavia in progetto un adattamento dello stesso algoritmo su base fonologica italiana.

Per la variabile 1.2. il messaggio offre un suggerimento, un sinonimo o una frase esemplificativa, che aiuti a identificare la parola non compresa. Tale aiuto può essere comunque richiamato, indipendentemente dalla parola battuta, premendo il tasto F2 nella fase di controllo.

Riportiamo ora qualche stringa d'esempio per quelle variabili del gruppo 2. non comprese in (r. 2):

«chissà» (r, 3):
elseif chi = «chissa»
risp = «C'è un accento in questa parola» (var. 2.6.)

«tagliare» (r. 5): *elseif tag = «taliare».or.tag= «talliare».or.tag. = «taiare*
risp= « Attento alla pronuncia della seconda sillaba» (var. 2.2)

«giorni» (r. 9)
elseif right(rtrim(gio), 1) <> «i».and.left (rtrim (gio), 5) = «giorn»
risp= «Attento: devi concordare la parola con «tutti i»(var. 2.4.)

«specchio» (r. 16):
elseif spe = «sbecchio»
risp = «Attento alla pronuncia: hai confuso la seconda lettera» (var. 2.3.)

La sintassi di programmazione utile per le nostre variabili si riduce quindi a pochi comandi fondamentali che, se opportunamente combinati, consentono una congrua individuazione di un buon numero di risposte.

Il rischio di messaggi del tutto incoerenti si ha praticamente solo nel caso di risposte inverosimili rispetto all'originale della canzone, limitando così il pericolo del cosiddetto *booh factor*, la tentazione dello studente di 'farsi beffe' del computer [6].

Un siffatto programma – qui illustrato solo in alcune sue parti – può apparire, a un primo esame, laborioso e per certi versi antieconomico. In effetti, se si considera che per l'analisi di tredici risposte si sono dovute inserire oltre 700 stringhe di programmazione e che per un esercizio che, tutto sommato, si esaurisce in una mezz'ora sono occorse parecchie ore di lavoro, è facile immaginare come ciò non rappresenti un buon incentivo per l'insegnante medio e digiuno di nozioni d'informatica a costruire programmi in proprio. Va detto tuttavia che buona parte del lavoro si riduce in fin dei conti a un'attività, per così dire, di copista: le istruzioni devono essere battute parecchie volte e si ripetono quasi sempre identiche o tutt'al più simili fra loro. Inoltre, il programma, con opportuni accorgimenti, può essere trasformato in un programma autore intelligente: riducendo al minimo e rendendo agevole la sintassi di programmazione, all'insegnante non resterà che il lavoro 'fondamentale' strettamente linguistico e didattico di pensare e inserire opportunamente l'analisi delle rispsote.

Nondimeno, il rapporto fra il lavoro necessario a costruire una lezione e il tempo di effettivo utilizzo di quest'ultima non sarà mai paritario nell'ambito del software didattico, almeno allo stato attuale degli sviluppi della ricerca informatica.

D'altra parte, tanto più complessa è l'impostazione e la struttura di un programma CALL, tanto più l'allievo avrà l'impressione che il computer 'reagisca' davvero alle sue risposte.

BIBLIOGRAFIA

Bibliografie CALL

U. O. H. Jung, *An International Bibliography of Computer-Assisted Language Learning with Annotations in German*, Peter Lang, Frankfurt/M.-Bern-New York-Paris 1988. La seconda edizione della bibliografia uscirà nel corso di quest'anno.

G. Lovere Di Maria e G. Porcelli, *Educazione linguistica assistita dal computer: una bibliografia*, in G. Mazzotti (a cura di), *Lingue, tecnologia e unione europea*, Marzorati, Settimo Milanese 1987, pp. 167-190.

Computer e insegnamento delle lingue

Ricordiamo innanzitutto il testo di uno degli studiosi che, in Italia, con maggiore impegno si è dedicato alla problematica CALL: G. Porcelli, *Computer e glottodidattica*, Liviana, Padova 1988. Quasi un classico nel suo genere si può considerare in Germania il seguente volumetto, giunto ormai alla seconda edizione: B. Rüschoff, *Fremdsprachenunterricht mit computergestützten Materialen. Didaktische Überlegungen und Beispiele*, Hueber, München 1986 (1982).

Ottimi consigli pratici sull'impiego dei programmi-autore troviamo in: M. Grüner e T. Hassert, *Computer in Unterricht. Voraussetzungen, Möglichkeiten, Grenzen*, Goethe-Institut, München 1991. (I capitoli 4 e 5 offrono una descrizione completa del programma *Storyboard - Textbaumeister* nella versione tedesca – corredata di riflessioni didattico-metodologiche e di esempi d'impiego. Completa il volume un'esauriente bibliografia).

Fra i lavori apparsi in lingua italiana ricordiamo inoltre:

E. Borello, M.C. Gozzoli, P. Hacht, A. Tollini, *Insegnare le lingue con il calcolatore*, Centro Scientifico Torinese, Torino 1987.

C. Marangon (a cura di), *Glottodidattica e tecnologie avanzate*, «Quaderni di ricerca C.L.I», Centro Linguistico Interfacoltà, Venezia 1991.

G. Porcelli (a cura di), *Il computer nella ricerca linguistica e in glottodidattica*, «Scuola e Lingue Moderne», 6 (1988) [numero monografico].

A. Rivella e R. Degl'Innocenti, *Programmare in Pascal. Applicazioni linguistiche*, Zanichelli, Bologna 1990.

Repertori del software CALL

Per un'analisi dei principali requisiti e per una classificazione dei programmi CALL: G. Porcelli, *I sussidi glottotecnologici avanzati*, «Scuola e Lingue Moderne» 6 (1988), pp. 206-215.

Molto ricca e periodicamente aggiornata e la *ReCALL Software Guide*, edita dal CTI Centre of Modern Languages, University of Hull (l'edizione più recente – agosto 1990 – prende in esame circa 300 programmi).

Si veda inoltre: Landesinstitut für Schule und Weiterbildung (a cura di), *Fremdsprachen Unterrichtssoftware – Nachweisliste*, Soest 1991, e I. Andreadou, *Software für den Fremdsprachenunterricht*, George Olms Verlag, Hildesheim-Zürich-New York 1987. Anche se in parte ormai abbondantemente superata (la raccolta si ferma al 1985; pochi i programmi per PC compatibili IBM allora disponibili), la bibliografia di Andreadou offre un'accurata analisi di ognuno dei ben 120 programmi presi in esame, con brevi descrizioni, caratteristiche didattiche e indicazioni sui destinatari. Completa il volume un esauriente repertorio di case editrici specializzate, produttori e distributori di software, nonché istituzioni scientifiche che si occupano di CALL.

Utile, anche se non specifico per il software CALL: A.I.C.A. Associazione Italiana per l'Informatica ed il Calcolo Automatico (a cura di), *Didamatica 91. Software didattico. 7° censimento nazionale*, Masson, Milano-Parigi-Barcellona-Bonn 1991.

Si ricorda infine che da alcuni anni è attiva a Genova, presso l'Istituto Tecnologie Didattiche del C.N.R., la Biblioteca del Software Didattico, dove è a disposizione di docenti e ricercatori

per la consultazione il software didattico italiano e straniero più significativo.

Riviste CALL

I periodici più qualificati nel settore sono il trimestrale «The CALICO Journal», edito da Computer Assisted Learning & Instructural Consortium, Duke University, USA, e, in Europa, «ReCALL. Journal of the CTI Centre of Modern Languages, University of Hull».

In Italia si occupano saltuariamente di CALL le riviste destinate al mondo della scuola: «Compuscuola», Gruppo Editoriale Jakson, «Scuola e informatica», Armando Editore, e «Informatica, Telematica e Scuola», McGraw-Hill Libri Italia.

Più orientate al mondo degli ipertesti e dei progetti multimediali: «Golem. Newsletter di tecnologie e progetti formativi», edito dell'Istituto di Psicologia del C.N.R., Università di Roma e «Multimedia»

Software CALL

Fra i programmi autore, ricordiamo quelli distribuiti da WIDA/Eurocentres Learning Service, tutti disponibili anche in versione italiana: *Parole sottosopra, Che scelgo?, Che sarà?, Che aggiungo?, Che corrisponde?, Parole e Storie, Il mio dizionario, Giochi di vocabolario,* e il citato *Da parola a parola*.

I ricercatori della Duke University hanno messo a punto un ottimo programma, studiato *ad hoc* per gli insegnanti di lingue, che consente di creare un'ampia tipologia di esercizi glottodidattici personalizzati dall'insegnante-autore (CALIS – *Computer Assisted Language Instruction System*).

Buona versalità e facilità d'uso sono i vantaggi di *Question Mark*, un sistema autore per la valutazione dell'apprendimento disponibile anche in versione italiana. Con esso si possono realizzare tutorials, test e questionari anche senza alcuna conoscenza d'informatica.

1 G. Wazel, *Der computerunterst utzte Sprachunterricht in der Sackgasse?*, «Bulletin CILA», 52 (1990), p. 9.

2 G. Wazel, *cit.*, p. 10.

3 *Dettato Canoro* è il frutto di una collaborazione con Gino Schiavinato del Centro Linguistico Interfacoltà dell'Università di Venezia, al quale si deve l'intero impianto tecnico del programma. È solo grazie alle sue soluzioni e ai suoi suggerimenti che il progetto ha potuto essere realizzato. Desidero ringraziare anche il prof. Federico Albano Leoni dell'Università di Napoli per gli utili consigli nella stesura di questo articolo.

4 Si veda la recente raccolta di L. Costamagna, *Cantare l'italiano. Materiali per l'apprendimento dell'italiano attraverso le canzoni*, Guerra, Perugia 1990. Peccato che nell'acclusa cassetta non siano registrate le canzoni con la loro musica originale, ma solo i testi parlati, sfruttati per un mero esercizio di imitazione. Per il tedesco è attivo già da alcuni anni un *pool* di insegnanti del Goethe-Institut autori di due belle raccolte di canzoni di vario genere, integrate con un apparato di simpatiche indicazioni per lo sfruttamento didattico: *pool-lifdu, Mein Gespräch, meine Lieder, Liedermacher im Deutschunterricht*, Langenscheidt, Berlin-Wien-Zürich-New York 1986 e *Heute hier, morgen dort! Lider, Chansons und Rockmusic im Deutschunterricht*, Langenscheidt, Berlin-Wien-Zürich-New York 1991.

5 G. Minardi, *Criteri di valutazione del software nella didattica delle lingue straniere*, in G. Mazzotti (a cura di), *Lingue, tecnologie e unione europea*, Marzorati, Milano 1987, p. 125.

6 «aspettato» (r. 2),«chissà» (r. 3), «lunghi» (r. 4), «tagliare» (r. 5), «d'oro» (r. 6), «gente» (r. 7), «d'oro» (r. 8), «giorni» (r. 9), «cuscino (r. 10), «ragazzino» (r. 11), «sottili» (r. 13), «bianchi» (r. 14), «pettinare» (r. 15), «specchio» (r. 16).

7 G. Porcelli, *Computer e glottodidattica*, Liviana, Padova, p. 13.